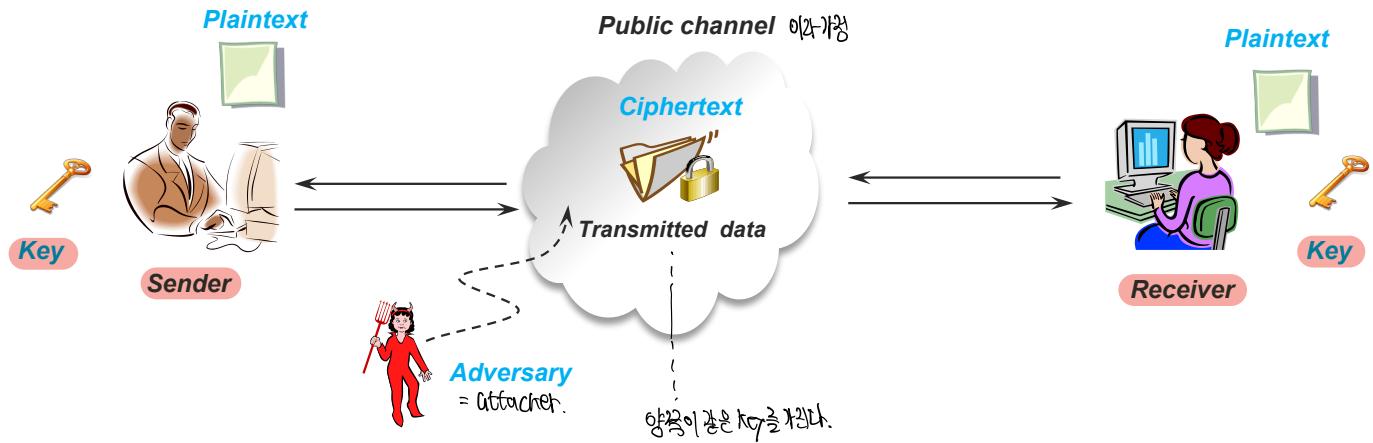


Classical Symmetric-key Ciphers

고급 암호
(대칭키 암호)

Jong Hwan Park

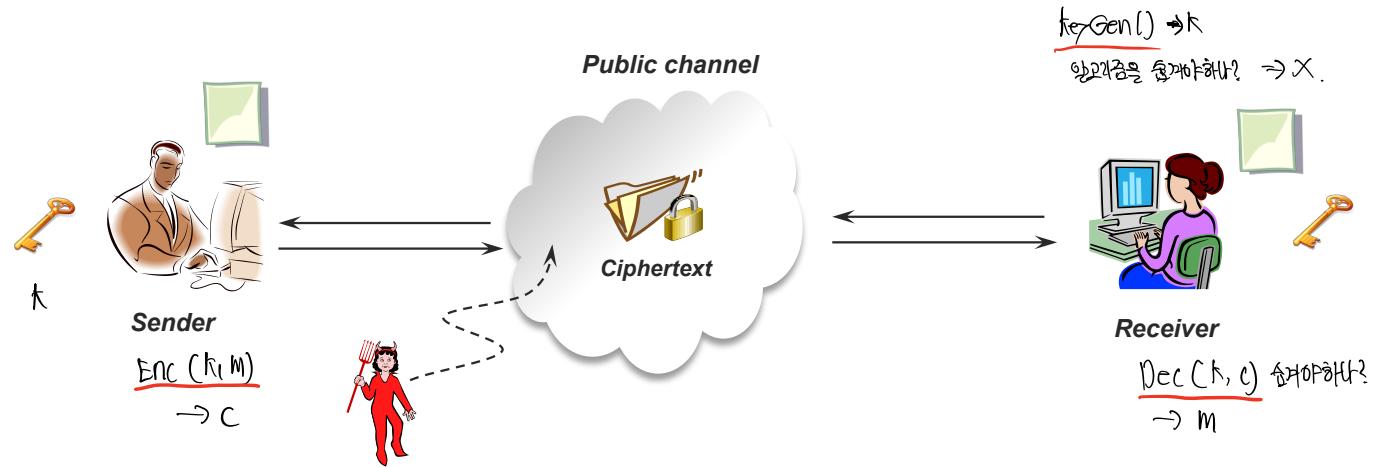
Symmetric-Key Encryption



- Definition of symmetric-key encryption
 - Key generation algorithm: $\text{Gen} \rightarrow$ a key $k \in KS$ (key space)
 - Encryption algorithm: $\text{Enc}(k, m) \rightarrow$ a ciphertext $C \in CS$ (ciphertext space)
 - Decryption algorithm: $\text{Dec}(k, C) \rightarrow$ a plaintext $m \in MS$ (message space)

Cf. Need to assume two parties initially share a key in a secure manner

Kerchoffs's Principle (1)



■ Clear to share the **key secretly and keep it secret** * 비밀로 전송하고 보관해야 한다.

■ Should the Enc and Dec algorithms be kept secret, too?

open.

■ Kerchoffs's opinion:

"The cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience."

키가 알고 있으면 키의 길이 = key length.

→ Demands that (1) security rely solely on the secrecy of the key, and (2) algorithms be made public

Kerchhoff's Principle (2)

- (1) Why?
 - Much easier to share and securely store a short key than a (longer) program
 - Much easier to change the key than the (longer) algorithm in case the key is exposed (or refreshed) 키를 간편하게 교체.
key 를 간편하게 refresh.
 - Much easier to use the same algorithm/program, but with different keys, in case many pairs of people are involved
 - Reverse engineering of the code poses a serious threat

- (2) Why?
 - Published designs undergo public scrutiny
 - Better for security flaws to be revealed by “ethical cryptographers”
 - Enable the establishment of standards

 Kerchhoff's Principle 원칙에 의해 암호화 알고리즘은 오픈해, 키가 공개되어 있어 안전성을 확보 (공개 대안화)

Caesar(shift) cipher (1)



Dec

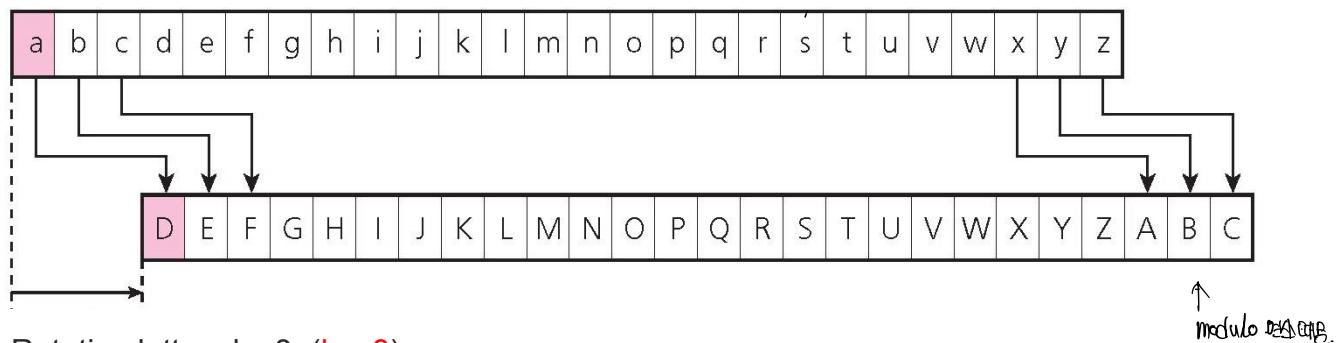


Ciphertext
(sent from Caesar)

Plaintext



Julius Caesar



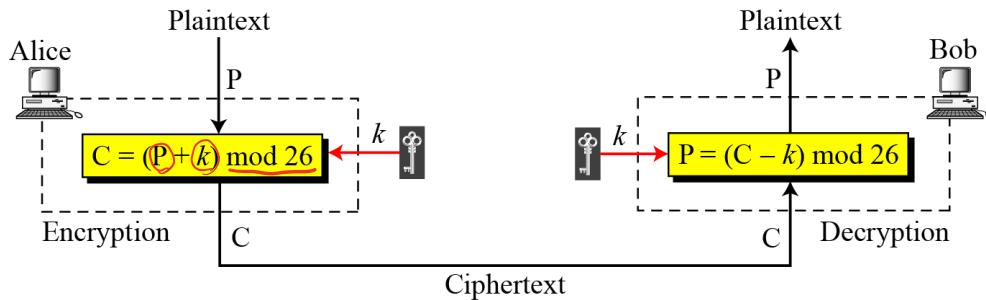
Caesar(shift) cipher (2)

- ## ■ Plaintext and ciphertext in $Z_{26} = \{0, 1, 2, \dots, 25\}$

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

$$\text{ex)} \quad 23 + 3 = 26 \equiv 0$$

- ## ■ Mathematical description



- ## ■ What is the problem?

- a key $\in \{0, 1, \dots, 25\} = Z_{26}$; brute-force attack or exhaustive search !!
key space \rightarrow 26 letters.
 - Need the **sufficient key space principle** $\|KS\| = 2^{128} \rightarrow 128\text{-bit security level}$

교훈 \Rightarrow key space는 충분히 크지 않다.

$$\|KS\| = 2^{128} \rightarrow 128\text{-bit security level}$$

128-bit ^{of} 디스 / 앤디스.

Mono-alphabetic Substitution Cipher

- Map each character to a different ciphertext character in an arbitrary manner

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	N	O	A	T	R	B	E	C	F	U	X	D	Q	G	Y	L	K	H	V	I	J	M	P	Z	S	W

↳ 키 테이블 mapping, / mapping table 을 사용하여 가짐 ↳ 키 테이블

- Example of encryption

Plaintext → this message is easy to encrypt but hard to find the key

Ciphertext → ICFVQRVVNEFVRNVSIYRGAHSLIOJICNHTIYBFGTICRXRS

- Security

- $|KS| = 26! = 26 \cdot 25 \cdot 24 \cdots 2 \cdot 1$ (approximately 2^{88}) $\doteq 2^{88}$
- Easy to break this scheme even if $|KS|$ is very large. Why?

Analysis of Mono-alphabetic Substitution Cipher (1)

■ Average letter frequencies for English-language text

Letter	Frequency	Letter	Frequency	Letter	Frequency	Letter	Frequency
E	12.7	H	6.1	W	2.3	K	0.08
T	9.1	R	6.0	F	2.2	J	0.02
A	8.2	D	4.3	G	2.0	Q	0.01
O	7.5	L	4.0	Y	2.0	X	0.01
I	7.0	C	2.8	P	1.9	Z	0.01
N	6.7	U	2.8	B	1.5		
S	6.3	M	2.4	V	1.0		

영어 평문 농자에 대한 빈도수.

■ Frequencies for diagrams and trigrams

Digram	TH, HE, IN, ER, AN, RE, ED, ON, ES, ST, EN, AT, TO, NT, HA, ND, OU, EA, NG, AS, OR, TI, IS, ET, IT, AR, TE, SE, HI, OF
Trigram	THE, ING, AND, HER, ERE, ENT, THA, NTH, WAS, ETH, FOR, DTH

Analysis of Mono-alphabetic Substitution Cipher (2)

- Ciphertext

MEYLGVIWAMEYOPINYZGWYEGMZRUUYPZAIXILGVSIZZMPGKK
DWOMEPGROEIWGPCEIPAMDKEYCIUYMGIFRWCEGLOPINYZHR
ZMPDNYWDWOGWITDWYSEDCEEIAFYYWMPIDWYAGTYPIKGLM
XFPIWCEHRZMMEYMEDWOMGQRYWCEUXMEDPZMQRGMEEYAP
ISDWOFICJILYSNICYZEYMGGJIPRWIWAIHRUNIWAHRZMUDZZYA
MEYFRWCEMRPWDWOPGRWAIOIDWSDMEIGWYMSGMEPYYEYH
RUNYARNFRMSDMEWGOPYIMYPZRCCYZZIOWIWAIOIDWEYMP
DYAILMYPMEYMWUNMDWOUGPZYKFRMIMKIZMEIAMGODTYD
MRNIWASIKJYAIISIXSDMEEDZWGZYDWMEYIDPZIXDWODIuzRPY
MEYXIPYZGRPDMDZYIZXMGAYZNDZYSEIMXGRCIWWGMOYM

Analysis of Mono-alphabetic Substitution Cipher (3)

- Frequency of characters in the ciphertext

I	47	G	27	C	12	F	7	V	2
Y	47	Z	27	S	11	L	6	B	0
M	45	P	26	N	10	H	5		
W	35	R	22	U	10	J	3		
E	33	A	17	K	8	T	3		
D	30	O	16	X	8	Q	2		

⇒ 암호문을 보면 평문의 빈도를 알 수 있다.

암호분석 : 암호문의 빈도와 평문의 빈도와 같은지.

Analysis of Mono-alphabetic Substitution Cipher (4)

- Substitution of the most frequent letter I or Y into e
- Assume that Y → e

T h

MEELGVIWAMEeOPINeZGWeEGMZRUUePZAIXILGVSI^TZMPGKKDW
OMEPGROEIWGPCeIPAMDKE^heCIUeMGIFRWCEGLOPINEZHRZMPD
NeWDWOGWITDW^heSEDCEEIAF^heewMPIDWeAGTePIKGLMXFPIWCEH
RZMMEEMEDWOMGQR^heWCEUXMEDPZMQRGMEEEAPISDWOFICJIL
eSNICEZEeMGGJIPRWIWAIHRUNIWAHRZMUDZZeAMEeFRWCEMRP
WDWOPGRWAIOIDWS^hDMEIGWeMSGMEPeEeHRUNEARNFRMSDME
WGOPeIMePZRCCeZZI^hOIDWTIWAIOIDWEeMPDeAILMePM^hEeMWUNM
DWOUGPZeKFRMIMKIZMEIAMGODTeDMRNIWASIKJeAISIXSDMEE
DZWGZeDW^hMEeIDPZIXDWODIUZRPeMEeXIPeZGRP^hDMDZeIZXMGA
eZNDZeSEIMXGRCIWWGMOeM

Analysis of Mono-alphabetic Substitution Cipher (5)

theLGVIWAtheOPINeZGWehGtZRUUePZAIXILGVSI~~Z~~tPGKKDW~~O~~thPG
ROhIWGPChIPAtDKKheCIUetGIFRWChGLOPINeZHRZtPDNeWDWOG
WITDWeShDChhIAFeeWtPIDWeAGTePIKGLtXFPIWChHRZt**the**thDW~~O~~t
GQReWChUXthDPZtQRGthheAPISDWOFICJILeSNICeZhetGGJIPRWIW
AIHRUNIWAHRZtUDZZeA**the**FRWCh~~t~~RPWDWOPGRWAIOIDWSDthIG
WetSGthPeeheHRUNeARNFRtSDthWGOPeItePZRCCeZZOIDWIWAIOID
WhetPDeAILteP**the**tWUNtDWOUGPZeKFRtItKIZthIAtGODTeDtRNIWASI
KJeAISIXSDthhDZWGZeDW**the**IDPZIXDWODIUZRP**the**XIPeZGRP~~D~~tDZ
eIZXtGAeZNDZeShItXGRCIWWGt**Oet**

thPee
⇒ th~~e~~e

Analysis of Mono-alphabetic Substitution Cipher (6)

theLoVanAthegraNeZonehotZRUUerZAaXaLoVSaZZtroKKingthroRghanorc
harAtiKKhecaUetoaFRnchoLgraNeZHRZtriNeningonaTineShichhaAFeenrai
neAoTeraKoLtXFranchHRZtthe thing to QRenchUXthirZtQRothheAraSingFac
JaLeSNaceZhetooJarRnanAaHRUNanAHRZtUiZZeAtheFRnchtRrningroRnA
againSithaonetSothreeheHRUNEARNFRtSithnogreaterZRcceZZagainanAagai
nhetrieAaLterthetnUNtingUorZeKFRtatKaZthaAtogiTeitRNaNASaKJeAaSaX
SithhiZnoZeintheairZaXingiaUZRretheXareZoRritiZeaZXtoAeZNiZeShatXo
Rcannotget

Shich
= which

Analysis of Mono-alphabetic Substitution Cipher (7)

thefoxandthegrapesonehotsummersdayafoxwasstroKKingthroughanorchard
tiKKhecametoaFunchofgrapesHustripingonaTinewhichhadFeentrainedoT
eraKoftyFranchHustthethingtoquenchmythirstquothedrawingFacJafewpac
eshetooJarunandaHumpandHustmissedtheFuncturningroundagainwithaone
twothreeheHumpedupFutwithnogreatersuccessagainandagainhetriedafterthe
tnmptingmorseKFutatKasthadtogiTeitupandwaKJedawaywithhisnoseinthea
irsayingiamsuretheyaresouritiseasytodespisewhatyoucannotget

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
I	F	C	A	Y	L	O	E	D	H	J	K	U	W	G	N	Q	P	Z	M	R	T	S	V	X	B

Analysis of Mono-alphabetic Substitution Cipher (8)

the fox and the grapes one hot summer's day a fox was strolling through an orchard till he came to a bunch of grapes just ripening on a vine which had been trained over a lofty branch. "Just to quench my thirst," quothe he. Drawing back a few paces, he took a run and a jump, and just missed the bunch. Turning round again with one, two, three, he jumped up, but with no greater success. Again and again he tried after the tempting morsel, but at last had to give it up, and walked away with his nose in the air, saying: "I am sure they are sour." It is easy to despise what you cannot get.

"The Fox and the Grapes"

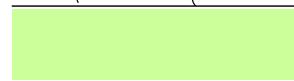
One hot summer's day, a Fox was strolling through an orchard till he came to a bunch of grapes just ripening on a vine which had been trained over a lofty branch. "Just to quench my thirst," quothe he. Drawing back a few paces, he took a run and a jump, and just missed the bunch. Turning round again with one, two, three, he jumped up, but with no greater success. Again and again he tried after the tempting morsel, but at last had to give it up, and walked away with his nose in the air, saying: "I am sure they are sour." It is easy to despise what you cannot get.

- Lesson:



- Any information about plaintexts should not be revealed from ciphertexts

~~보기~~: 평문에 대한 정보(천장)이 그대로 암호화에 보호해줄 때서 정보 유출을 막는다.



P: [E: C]
암호화

Vigenere (poly-alphabetic shift) Cipher

- To remove character frequencies in a ciphertext
- How it works:

$$\text{key size} = 26 \cdot 26 \cdots \times 26) = 26^d$$

- Key = (k_1, k_2, \dots, k_d) (repeated as many times as needed)

1. Key size는 무한하다
2. 평문의 차이가 암호문으로
오지 않음.

- $E_K(x) \equiv (x_1 + k_1, x_2 + k_2, \dots, x_d + k_d) / (Caesar\ cipher)$
 $D_K(x) \equiv (c_1 - k_1, c_2 - k_2, \dots, c_d - k_d) / (Poly)$

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

- Example of encryption: key = (PASCAL)=(15, 0, 18, 2, 0, 11)

Shift + Mod	Plaintext:	s h e i s l	i s t e n i	n g
	P's values:	18 07 04 08 18 11	08 18 19 04 13 08	13 06
	Key stream:	15 00 18 02 00 11	15 00 18 02 00 11	15 00
	C's values:	07 07 22 10 18 22	23 18 11 6 13 19	02 06
	Ciphertext:	H H W K S W	X S L G N T	C G

⇒ 동일문자 ⇒ 다음문자 대입 Mono-alphabetic cipher의 원리와 비슷.

Analysis of Vigenere Cipher (1)

■ Step 1

- Find the length of the key

- Kasiski test:

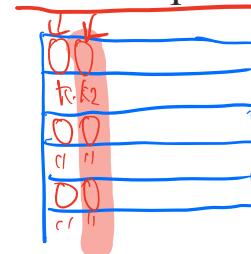
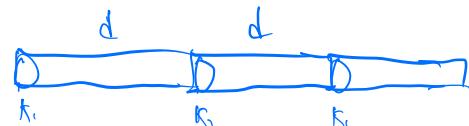
- Search for repeated segments, of at least 3 characters, in the ciphertext
- Find the distance d between two of segments
- Using more segments, find d_1, d_2, \dots, d_n
- **Key length** could be one of the factors of $\gcd(d_1, d_2, \dots, d_n)$
⇒ 키의 길이를 찾을 수 있다.

Ex) CH2 - ... | CH2

■ Step 2 (if the key length m is found)

- Divide the ciphertext into m pieces

- Apply the frequency analysis to each piece used in mono-alphabetic substitution cipher



Analysis of Vigenere Cipher (2)

- Example of a ciphertext:

LIOMWGEGGDVWGHHCQUCRHRWAGWIOWQLKGZETKKMEVLWPCZVGTH-
VTSGXQOVGCSVETQLTJSUMVVVEUVLXEWSLGFZMVVWLGYHCUSWXQH-
KVGSHHEEVFLCFDGVSUMPHKIRZDMPHHBVWWJWIXGFWLTSHGJOUEEHH-
VUCFVGOWICQLTJSUXGLW

- Kasiski test yields the results:

<i>String</i>	<i>First Index</i>	<i>Second Index</i>	<i>Difference</i>
JSU	68	168	100
SUM	69	117	48
VWV	72	132	60
MPH	119	127	8

- $\gcd(100, 48, 60, 8) = 4$ (which is a key length)

Analysis of Vigenere Cipher (3)

- Divide the ciphertext into 4 pieces:

C1: LWGWCRAOKTEPGTQCTJVUEGVGUQGECVPRPVJGTJEUGCJG

1

P1: jueuapymircneroarhtsthihytrahcieixsthcarrehe

14

C2: IGGGQHGWGKVCTSOSQS WVWFVYSHSVF SHZHWWF SOHCOQSL

14

P2: usssctsis who feaeceihcetes oecatnpntherhctecex

3

C3: OFDHURWQZKLZHGVVLUVLSZWHWKHF DUKDHVIWHUHF WL UW

3

P3: lcaerotnwhi wed ssirsi irhkete hretl t i ideat rai rt

3

C4: MEVHCWILEMVVVXGETMEXMLCXVELGMIMBWXLGEVVITX

4

P4: i ardysehaisrrt capiafpwtethecarhaesft erectpt

- Key = (2, 14, 3, 4) = (c o d e)

- The resulting plaintext makes sense:

Julius Caesar used a cryptosystem in his wars, which is now referred to as Caesar cipher. It is an additive cipher with the key set to three. Each character in the plaintext is shifted three characters to create ciphertext.

Important Lessons

- Sufficient key space principle
- $|KS|$ vs. $|Ciphertext|$ required for analysis සුංඝාල ප්‍රමාණ.

 - $|Key Space|$: Caesar < Mono-alphabetic sub. < Vigenere
 - $|Ciphertext|$: Caesar < Mono-alphabetic sub. < Vigenere

- Designing secure ciphers is a hard task
 - All historical ciphers can be completely broken
 - Even far more complex schemes, such as the German Enigma

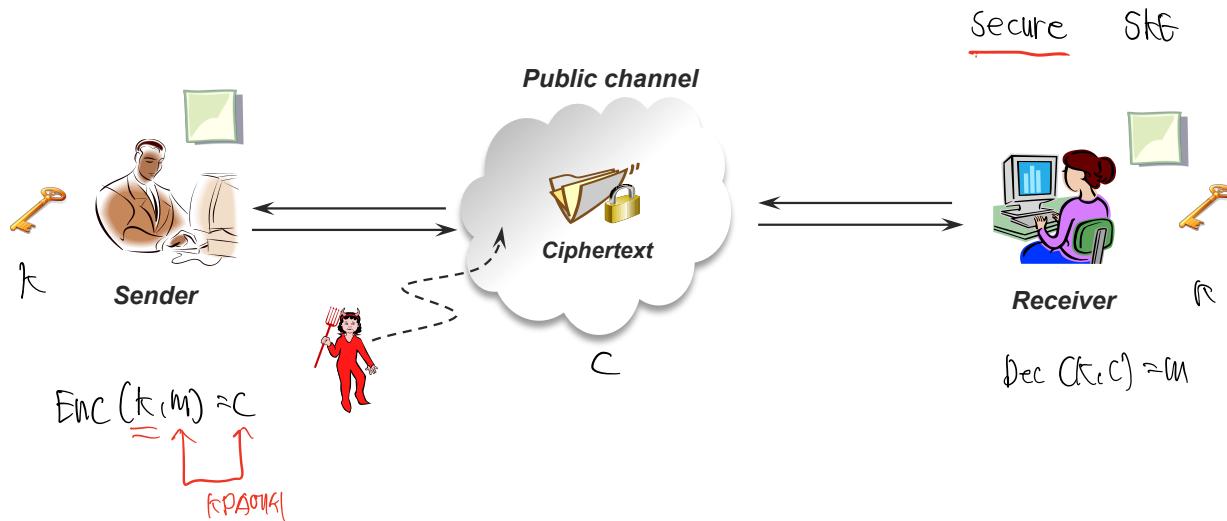
$$|K| = 2^{128} \rightarrow \text{large key}$$
$$|K'| = 2^{128} \rightarrow \text{large key}$$

update



Attack Scenarios (1)

인천성을 공격해보도록 하자.



■ Ciphertext-Only Attack (COA)

- Just observe $\{C_i\}$ and aim to determine the underlying m corresponding to some target ciphertext

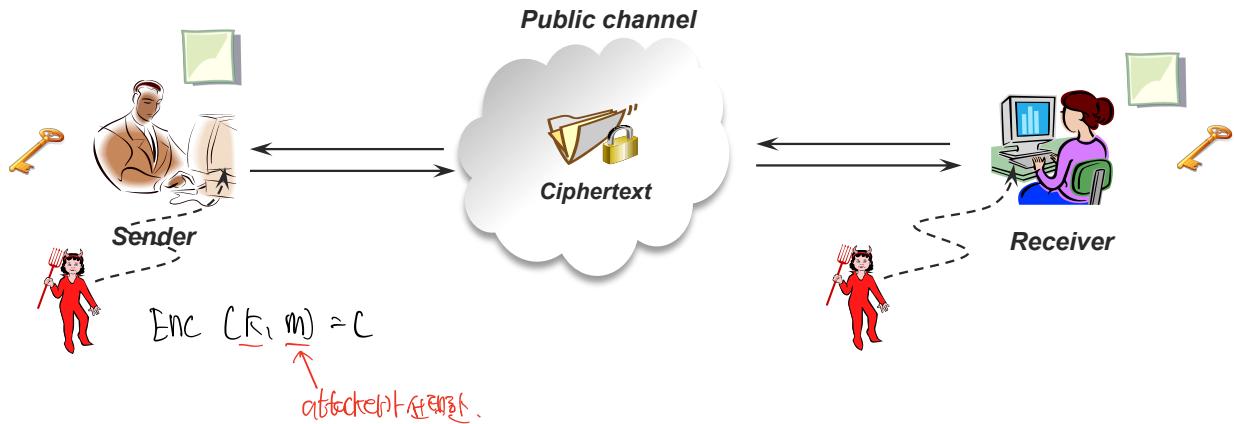
공개복호화 알고리즘을 통해 키를 찾는 방식이기는 하지만 몇몇 암호는
마지막에 대한 풀이를 찾을 수 있다.

■ Known-Plaintext Attack (KPA)

- Learn $\{m_i, C_i\}$ under the same key and aim to determine the m corresponding to some target ciphertext

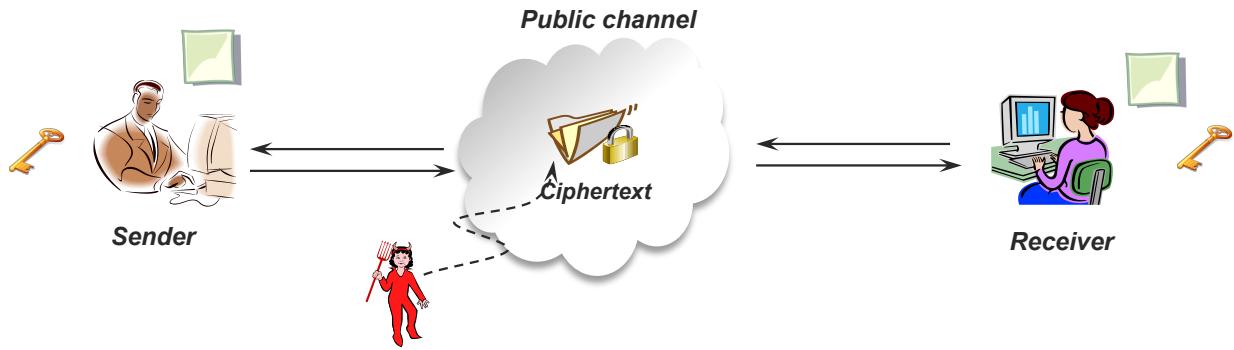
공개복호화 알고리즘을 통해 키를 찾는 방식은 가능하지만, 모든 것은 가능.

Attack Scenarios (2)



- **Chosen-Plaintext Attack (CPA)** 암호해독자는 텍스트를 선택해 키를 가진 상태에 암호화를 한다.
 - Have the ability to obtain $\{C_i\}$ of plaintexts $\{m_i\}$ of its choice
 - Aim to determine the m corresponding to some target ciphertext
- **Chosen-Ciphertext Attack (CCA)** 1. 텍스트 선택해 대응하는 암호문 찾기 (CPA)
2. 키의 Dec 과정에서 키를 만들 때는 암호문 선택 허용
- Have the ability to obtain $\{m_i\}$ of ciphertexts $\{C_i\}$ of its choice
 - Aim to determine the m corresponding to some target ciphertext

Security & Efficiency



- Attack hardness: COA < KPA < CPA < **CCA**
- Not always the case that an encryption scheme secure against CCA should be used
 - May be less efficient than a scheme secure against *weaker* attacks
 - Tend to follow the fact that:



Two Approaches to Modern Cryptography

■ How can one design cryptographic primitives?

○ Cryptanalysis-driven design 부록 → 허설과 보안

- Problem → proposed solution → bug → revised solution → ...
→ Implement → Bug → ... 

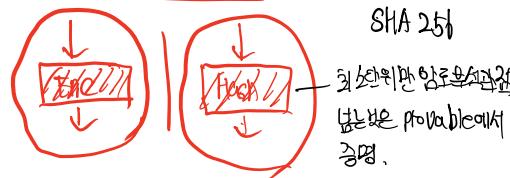
AES
↓
SHA
↓

- Symmetric-key encryption, MAC(Message Authentication Code), hash function, ... AES

최소한의 function들을 암호화설계에서 실제 → 흐름 → 보안 ⇒ 외장공 (최소단위)

문제 : 기밀성보증 시도는 악의적(Attacker) → 내부적(Internal) 깨끗하다.

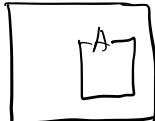
SHA 256



○ Provable-security approach

- Problem → Definition → Scheme → Reduction → Implement → Done
- Based on a complexity assumption 증명.
 - Infeasible to solve it in polynomial-time 인증성 증명.
 - {Diffie-Hellman, RSA, Discrete logarithm} problem ...
- Public-key encryption, digital signature, key agreement, authentication, ...

Security
model



설명 : 거의 모든 고급은 Provable-Security 설계에서 증명이면 기법을 사용.
⇒ 현대암호학 증명화 설계학자.

Basic Principle of Provable Security

- To solve any cryptographic problem
 - P1 - the formulation of a precise definition of security
(i.e., attack model) A
 - P2 - when the security relies on a complexity assumption,
this assumption must be precisely stated
보안assumption.
 - P3 - the security should be proven rigorously
under the definition of (P1) and the assumption of (P2)
대수학적 증명.

■ Security proof in provable security

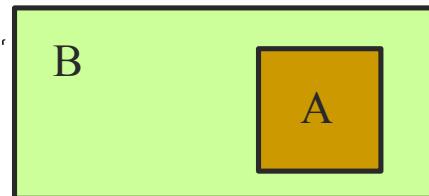
If the assumption X holds, a cryptographic scheme Y is secure (against a certain attack)

증명되어 보안assumption.

이것을 provable security로 부르기도 한다.

$X \rightarrow Y$

CPA or OA



평균 험성을 명.

→ 이를 알고리즘으로 가정

Q & A